Simplification of CFG lecture 18

25/05/20

1/ Remove useless symbols or productions

2/ Remove null productions $A \to \epsilon$

4) Remove recursion.

3) Remove unit productions.

Remove useless symbols

Algo Stage 1

① Make a table with OV, NV and P'.
Initialize OV with ϕ.

② Add those productions to P, that derive terminals only.

③ Add the variables to NV set.

④ Start a fresh row by taking NV to OV column and add productions of the form $A \to Y$ where $Y = (OV \cup T)$

When OV = NV stop.

Stage 2

1) Obtain the set of variables and terminals that are reachable from the start symbol

2) Productions which are not used are useless.

<u>Algo</u>

<u>stage 1</u>

$ov = \phi$

$nv = ov \cup \{A \mid A \to y \text{ and } y \in (ov \cup T)^*\}$

while ($ov \neq nv$)

{

$ov = nv;$

$nv = ov \cup \{A \mid A \to y \text{ and } y \in (ov \cup T)^*\}$

}

$V_1 = ov$

<u>Stage 2</u>

$V_1' = \{S\}$

for each A in V_1

If $A \to \alpha$ then

Add the variable in α to V'

Add the terminals in α to P'

end if

end for.

Using this algorithm all those symbols (terminals and variables) that are not reachable from the start symbol are eliminated.

Q/ Eliminate useless symbols in the G (3)

$S \to aA \mid bB$

$A \to aA \mid a$

$B \to bB$

$D \to ab \mid Ea$

$E \to aC \mid d$

OV	NV	Productions
ϕ	A, D, E	$A \to a$ $D \to ab$ $E \to d$
A, D, E	A, D, E, S	$S \to aA$ $A \to aA$ $D \to Ea$
A, D, E, S	A, D, E, S	—

Resulting grammar $G_1 = (V_1, T_1, P_1, S)$

$V_1 = \{A, D, E, S\}$

$T_1 = \{a, b, d\}$

$P_1 = \{ A \to a \mid aA$

$D \to ab \mid Ea$

$E \to d$

$S \to aA$

$\}$

S is the start symbol

stage 2 :→ obtain symbols that are reachable from S

P_1'	T_1	V_1
-	-	S
$S \to aA$	a	S, A
$A \to a \mid aA$	a	S, A

Resulting grammar $G' = (V', T', P', S)$

$V' = \{S, A\}$

$T' = \{a\}$

$P' = \{$
$S \to aA$
$A \to a \mid aA$
$\}$

S is the start symbol

Q/ Simplify the grammar

$S \to aA \mid a \mid Bb \mid cC$

$A \to aB$

$B \to a \mid Aa$

$C \to cCD$

$D \to ddd$

stage 1

OV	NV	Productions
ϕ	S, B, D	S → a B → a D → ddd
S, B, D	S, B, D, A	S → Bb A → aB
S, B, D, A	S, B, D, A	S → aA B → Aa

$G_1 = (V_1, T_1, P_1, S)$

$V_1 = \{ S, B, D, A \}$

$T_1 = \{ a, b, d \}$

$P_1 = \{ S \to a \mid aA \mid Bb$

$B \to a \mid Aa$

$D \to ddd$

$A \to aB$

$\}$

S is the start symbol.

stage 2

P'	T'	V'
S → a | Bb | aA	a, b	S, A, B
A → aB	a, b	S, A, B
B → a | Aa	a, b	S, A, B

$V' = \{S, A, B\}$ $T' = a, b$, $P' = \{ S \to a \mid Bb \mid aA$, $A \to aB$, $B \to a \mid Aa \}$

S is start.

Eliminating ε- productions

(a) A production of the form $A \to \varepsilon$ is undesirable in a CFG, unless an empty string is derived from the start symbol.

(b) Suppose the language produced from a Grammar does not derive empty string and the grammar consists of ε - productions, such ε - productions can be removed.

Q/ $S \to ABCa \mid bD$
$A \to BC \mid b$
$B \to b \mid \varepsilon$
$C \to c \mid \varepsilon$
$D \to d$.

$B \to \varepsilon$
$C \to \varepsilon$ } ~~direct~~
$A \to BC$
nullable.

OV	nV	Production
ϕ	B, C	$B \to \varepsilon$ $C \to \varepsilon$
B, C	B, C, A	$A \to BC$
B, C, A	B, C, A	—

$V = \{A, B, C\}$.

Productions	Resulting production
$S \to ABCa$	$S \to ABCa \mid BCa \mid ACa \mid ABa \mid Ca \mid Aa \mid Ba \mid a$.
$S \to bD$	$S \to bD$
$A \to BC \mid b$	$A \to BC \mid B \mid C \mid b$
$B \to b \mid \varepsilon$	$B \to b$
$C \to c \mid \varepsilon$	$C \to c$
$D \to d$	$D \to d$.

find G~~~~

Algorithm

Step 1 → Obtain the set of nullable variables from the grammar.

OV	NV	Productions

$OV = \phi$

$nV = \{ A \mid A \rightarrow \varepsilon \}$

while $(OV \ != nV)$

{

$OV = nV$

$nV = OV \cup \{ A \mid A \rightarrow \alpha$ and $\alpha \in OV^* \}$

}

$V = OV$

— set V contains only the nullable variables.

Step 2. Construction of productions P.

$A \rightarrow X_1 X_2 \ldots X_n, \; n \geq 1$

where $X_i \in (V \cup T)$

Take all possible combinations of nullable variables and replace the nullable variables with ϵ one by one and add the resulting productions to P'. (8)

If the given production is not an ϵ-production, add it to P'.

The resulting grammar generates the same language as generated by G without ϵ

$G = (V', T', P', S)$

$V' = \{S, A, B, C, D\}$

$T' = \{a, b, c, d\}$

$P' = \{ S \rightarrow ABCa \mid BCa \mid ACa \mid ABa \mid Ca \mid Ba \mid Aa \mid a$

$A \rightarrow BC \mid B \mid C \mid b$

$B \rightarrow b$

$C \rightarrow c$

$D \rightarrow d$

$\}$

S is the start symbol.

8) $S \to BAAB$

$A \to 0A2 \mid 2A0 \mid \epsilon$

$B \to AB \mid 1B \mid \epsilon$

OV	nV	Productions
ϕ	A, B	$A \to \epsilon$ $B \to \epsilon$
A, B	A, B, S	$S \to BAAB$
A, B, S	A, B, S	–

$V = \{S, A, B\}$

Productions	Resulting productions
$S \to BAAB$	$S \to BAAB \mid AAB \mid BAB \mid BAA \mid AB \mid BB \mid BA \mid AA \mid A \mid B$
$A \to 0A2$	$A \to 0A2 \mid 02$
$A \to 2A0$	$A \to 2A0 \mid 20$
$B \to AB$	$B \to AB \mid B \mid A$
$B \to 1B$	$B \to 1B \mid 1$

$G = (V', T', P', S)$

$V = \{S, A, B\}$

$T = \{0, 1, 2\}$

$P = \begin{cases} S \to BAAB \mid A \\ A \to 0A2 \mid 2A0 \mid 02 \mid 20 \\ B \to AB \mid B \mid A \mid 1B \mid 1 \end{cases}$

Eliminating Unit Productions

$A \rightarrow B \rightarrow$ unit production, where RHS and LHS of the production contain only one variable.

$A \rightarrow B$
$B \rightarrow aB \mid b$

$B \rightarrow aB \mid b$ are non-unit productions

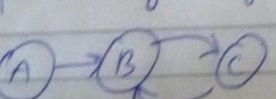
B is generated from A,
∴ whatever is generated by B can be generated by A also.

∴ $A \rightarrow aB \mid b$ is correct
and $A \rightarrow B$ is deleted.

Procedure

1. Remove the production of the form $A \rightarrow B$
2. Add all non-unit productions to P_1.
3. For each variable A find all the variables B such that
   $A \overset{*}{\Rightarrow} B$
   i.e, in the derivation process from A, if we encounter only one variable in a sentential form say B (no terminals should be there), obtain all such variables.
4. Obtain a dependency graph.

$A \rightarrow B$
$B \rightarrow C$
$C \rightarrow B$



i.e, C can be ~~for~~ generated from A.

$S \to AB$
$A \to a$
$B \to C \mid b$
$C \to D$
$D \to E \mid bC$
$E \to d \mid Ab$.

Sol: Non-unit productions of the grammar G are

$S \to AB$
$A \to a$
$B \to b$
$D \to bC$
$E \to d \mid Ab$

Unit productions are

$B \to C$
$C \to D$
$D \to E$

DG is

$B \to C \to D \to E$

$D \to E$ and all non-unit productions from E can also be generated from D.

$E \to d \mid Ab \implies D \to d \mid Ab$.

$\therefore D \to bC \mid d \mid Ab$.

$C \to E \implies C \to d \mid Ab$

But $C \to D \implies C \to bC \mid d \mid Ab$.

$B \to C$ so all put together

$B \to b \mid d \mid Ab \mid bC$

$\therefore$ $G = (V, T, P, S)$

$V = \{A, B, C, D, E\}$

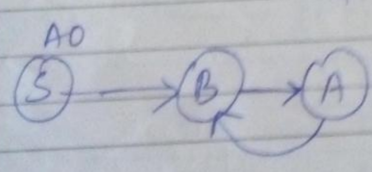
$T \to \{a, b, d\}$

$P = \{$
$S \to AB$
$A \to a$
$B \to b \mid d \mid Ab \mid bC$
$C \to bC \mid d \mid Ab$
$D \to bC \mid d \mid Ab$
$E \to d \mid Ab$
$\}$

S is the start symbol

Q/

$S \to A0 \mid B$
$B \to A \mid 11$
$A \to 0 \mid 12 \mid B$



$S \to B$
$B \to A$
$A \to B$

$A \to 0 \mid 12 \mid 11$
$B \to 11 \mid 0 \mid 12$
$S \to A0 \mid 11 \mid 0 \mid 12$